Provide a report on your findings from the pcap file and outline what processes / the steps you followed to achieve this. Here are each of your sub-tasks with additional instructions. Please record your findings under each sub-task title.

Sub-task 1:

- *anz-logo.jpg and bank-card.jpg are two images that show up in the users network traffic.*
- *Extract these images from the pcap file and attach them to your report.*

To find both images, I followed the same process: I filtered the packet capture for http traffic to confirm the images. Then I selected **File** then clicked **Extract Objects** from the dropdown. From the second dropdown I selected **HTTP** to only receive objects from http traffic. To know what objects to extract, I referred to the packet captures to select the correct one that responded to the GET request for these images.

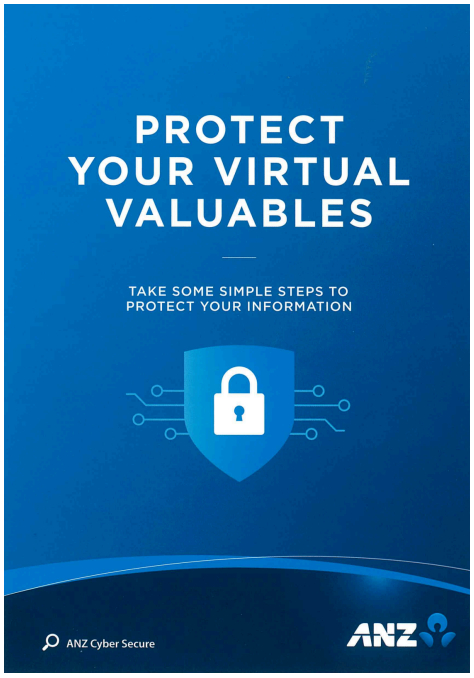
The left image is of anz-logo.jpg. The right image is of bank-card.jpg (resized).

Sub-task 2:

- *The network traffic for the images "ANZ1.jpg" and "ANZ2.jpg" is more than it appears.*
- *Extract the images, include them and mention what is different about them in your report.*

I followed the same process to extract these two images as I did in sub-task 1: from the menu I selected **File** > **Extract Objects** > **HTTP** to see the list of objects in the HTTP traffic. I downloaded the corresponding objects from the packet that responded to the GET request for these images.

The image on the left is of ANZ1.jpg (resized) and the image on the right is of ANZ2.jpg (resized).



To check if there is anything hidden, I filtered the packet for HTTP steam then viewed the TCP stream of the response to the get request of both images and saw a hidden message written at the end.
ANZ1.jpg message was “You've found a hidden message in this file! Include it in your write up.”
ANZ2,jpg message was “You've found the hidden message! Images are sometimes more than they appear.”

Sub-task 3:

- *The user downloaded a suspicious document called "how-to-commit-crimes.docx"*
- *Find the contents of this file and include it in your report.*
- *Actions*
    - *so can do extract object but can't open in VM so follow TCP stream*
    - *Contents*
        - *Step 1: Find target*
        - *Step 2: Hack them*
        - *This is a suspicious document*

While I was able to extract the document following the same process as in previous sub-tasks, I was unable to open it so I used another method. I filtered the packet for HTTP steam then viewed the TCP stream of the http get request for the document.

The image below is the TCP steam of the request with the document message being;
~~~
Step 1: Find target
Step 2: Hack them

This is a suspicious document.
~~~

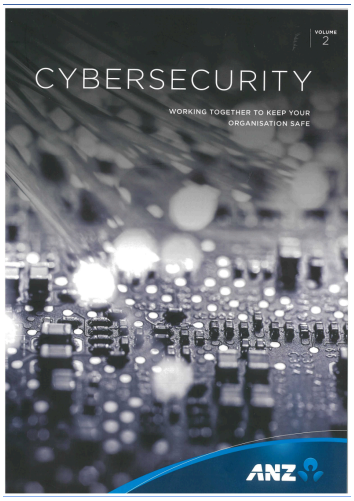
```
GET /how-to-commit-crimes.docx HTTP/1.1
Host: localhost:8000
Connection: keep-alive
Sec-Fetch-Site: same-origin
User-Agent: Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/76.0.3809.100 Safari/537.36
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9

HTTP/1.1 200 OK
Date: Fri, 16 Aug 2019 00:48:17 GMT
Server: Apache/2.4.6 (CentOS)
Last-Modified: Mon, 05 Aug 2019 02:23:32 GMT
ETag: "46-58f5564f85059"
Accept-Ranges: bytes
Content-Length: 70
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/vnd.openxmlformats-officedocument.wordprocessingml.document

Step 1: Find target
Step 2: Hack them

This is a suspicious document.
```

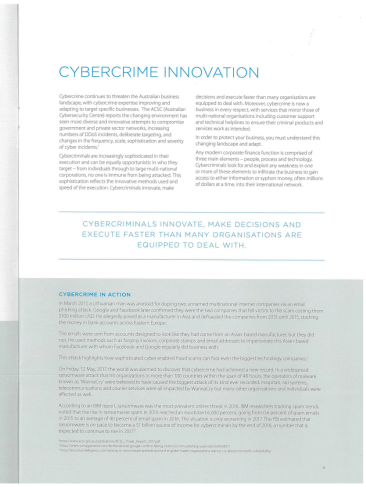
Sub-task 4:

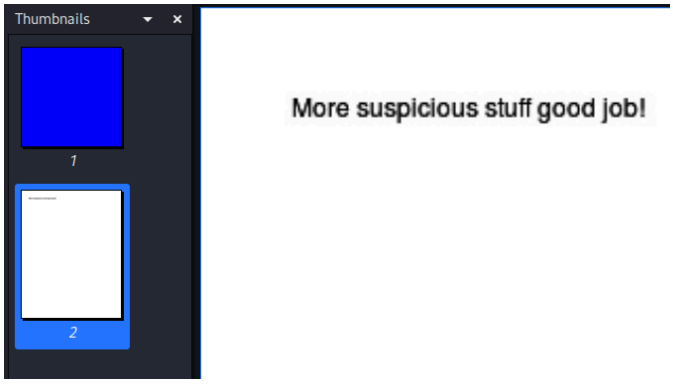
- *The user accessed 3 pdf documents: ANZ_Document.pdf, ANZ_Document2.pdf, evil.pdf*
- *Extract and view these documents. Include images of them in your report.*
- *Actions:*
    - *extract objects again*

I followed the same process to extract these two images as I did in sub-task 1: from the menu I selected **File** > **Extract Objects** > **HTTP** to see the list of objects in the HTTP traffic. I downloaded the corresponding objects from the packet that responded to the GET request for these documents.

The top left imageis of ANZ_Document.pdf (resized) and the top right image is of ANZ_Document2.pdf (resized). The last image is of evil.pdf (resized).







Sub-task 5:

- *The user also accessed a file called "hiddenmessage2.txt"*
- *What is the contents of this file? Include it in your report*

After extracting the file, I used the **xxd** command to see that the file type is JFIF instead of TXT. I edited the file extension to be .jfif, resulting in the image (resized) below being displayed.

Sub-task 6:

- *The user accessed an image called "atm-image.jpg"*
- *Identify what is different about this traffic and include everything in your report.*
- *Actions*
    - *Extract objects*
    - *This is not a file but a real photo of two atms*
- *another;*
    - *check tcp stream and saw two images*
    - *extracted object and saw first image below*
    - *went to cyberchef to extract files*
    - *output was 2 files*
    - *second is of robbery*

After extracting the image, I was able to open it without issue. This is the first image (resized) shown on the left.

While there did not seem to be any issue, I went to CyberChef to see if there was anything hidden. I placed the first image as the input and selected the option to extract objects. There were 2 image files shown. The first was the same as the extracted image while the second was a new image. The second image (resized) is shown on the right.

Sub-task 7:

- *The network traffic shows that the user accessed the image "broken.png"*
- *Extract and include the image in your report.*

After extracting the image, I was unable to open the image. I viewed the TCP stream of the GET request for this image and saw it was encoded in base64. I used CyberChef to decrypt the decoded and render image.

Below is the unencrypted image (resize) of broken.png. It is a transparent png file.

Sub-task 8:

- *The user accessed one more document called securepdf.pdf*
- *Access this document include an image of the pdf in your report. Detail the steps to access it.*

After extracting the file, I was unable to open the file and received a warning that it was a zip file. I viewed the TCP stream of the GET request of the file to confirm it was a zip file and found a hidden message:
~~~

Password is “secure”
~~~

I changed the file extension to .zip, and used the password provided. There was a pdf file inside called rawpdf.pdf.

Below is the two pages contained in the file rawpdf.pdf.